DIVIDING AND CONQUERING CROSS-MODAL RECIPE RETRIEVAL: FROM NEAREST NEIGHBOURS BASELINES TO SOTA

A PREPRINT

Mikhail Fain*
Cookpad Ltd.
mikhail-fain@cookpad.com

Andrey Ponikar*
Cookpad Ltd.
andrey-ponikar@cookpad.com

Ryan Fox
Cookpad Ltd.
ryan-fox@cookpad.com

Danushka Bollegala
University of Liverpool
danushka@liverpool.ac.uk

December 2, 2019

ABSTRACT

We propose a novel non-parametric method for cross-modal retrieval which is applied on top of precomputed image and text embeddings. By combining our method with standard approaches for building image and text encoders, trained independently with a self-supervised classification objective, we create a baseline model which outperforms most existing methods on a challenging image-to-recipe task. We also use our method for comparing image and text encoders trained using different modern approaches, thus addressing the issues hindering the developments of novel methods for cross-modal recipe retrieval. We demonstrate how to use the insights from model comparison and extend our baseline model with standard triplet loss that improves SoTA on the Recipe1M dataset by a large margin, while using only precomputed features and with much less complexity than existing methods.

Keywords Cross-Modal Recipe Retrieval · Baselines · Nearest Neighbours · Self-Supervised Learning

1 Introduction

In this work we are exploring the problem of cross-modal recipe retrieval between food images and textual cooking recipes. A solution to this problem has a number of applications, such as searching for the correct recipe using a photo [1], automatically determining the number of calories in a dish [2] and improving the performance of various recipe recommendation and ranking systems [3]. Note, that we only focus on image-to-textual-recipe retrieval rather than textual-recipe-to-image retrieval for the sake of clarity. This task involves searching for an exact matching text of the recipe given its image among candidate textual recipes from a held-out test set.

The problem of recipe retrieval is challenging due to the diverse nature of food images and the subtle differences between recipes. For example, the photos of dishes made by following the same recipe could look completely different from each other, and very similar dish photos could be associated with very different ingredients and procedures[2].

After the release of the Recipe1M dataset [4] containing diverse food images and recipes, cross-modal recipe retrieval became one of the standard benchmarks for image-to-document retrieval tasks, with numerous improvements made on the original model [4]. Despite this progress, we observe a few issues hindering further development of the new methods in this space: namely, the lack of strong baselines and the complexity of identifying strengths and weaknesses of individual model components across methods.

*Equal contribution
[2]As an example, consider visually distinguishing different types of creamy soups from each other using only photos

Problem 1: Lack of strong baselines. In the seminal paper on cross-modal recipe retrieval, Salvador *et al.* [4] presented a baseline using one of the strongest statistical models for learning joint embeddings, Canonical Correlation Analysis (CCA) [5]. This model achieved the top-1 accuracy of 14.0 on a test set of 1,000 recipes. A deep learning model presented in the same paper almost doubled the top-1 accuracy reaching 25.6. By 2019, after two years of steady improvements [6, 7, 8, 9], this metric was doubled again to reach 51.8 [10]. While this could reflect the improvements in retrieval methods, it could also be due to the weakness of the original baseline. If the latter is the case, a strong and simple baseline for cross-modal retrieval could facilitate faster scientific progress and put the previously reported results into perspective.

Problem 2: Complexity of identifying different models' components strengths and weaknesses. Cross-modal recipe retrieval models are very complex and have many interacting parts, thus the performance gains are not easy to attribute to improvements in a particular part such as an image encoder, a text encoder or a modality alignment procedure. This makes it extremely hard to make an educated guess about which ideas from one model could be re-used or combined with another model, as ablation studies only address performance gains within the same method. The prior work on cross-modal recipe retrieval [7, 8, 9, 10] mitigates this issue by using an image and textual recipe processing pipeline that is similar to the one introduced by Salvador *et al.* [4]. However, the setups still sufficiently differ to prevent a fair comparison [7, 8]. In addition, being tied to specific architectures presents a danger of the new methods being tailored to them and makes it especially hard to drive improvements in encoding recipe images and textual recipes.

Our paper contributes to addressing these two problems in the following way:

1. We propose a novel non-parametric method for cross-modal retrieval, **C**ross-Modal **k**-**N**earest-**N**eighbours (**CkNN**), which is applied on top of precomputed image and text embeddings. It is based on using k-Nearest-Neighbours (kNN) to search through the training data and across different modalities using the correspondences available in the training set.
2. We create a baseline model for a challenging image-to-textual-recipe retrieval task by combining CkNN with standard approaches for independently representing images and text using a self-supervised classification objective. This baseline outperforms most existing methods, addressing **Problem 1**.
3. Since CkNN has no parameters and depends directly on different modalities' distance measures, we use it for comparing image and text encoders trained with different approaches, addressing **Problem 2**.
4. We demonstrate how to use the insights from encoder comparison and extend our baseline model with triplet loss that improves SoTA by a large margin while using only precomputed embeddings and with much less complexity than existing approaches.

We hope that our approach for comparing different model's encoders and creating strong but simple baselines would encourage further development of advanced end-to-end methods.

2 Related Work

The problem of cross-modal retrieval has been researched extensively in the Computer Vision community. The majority of the recent solutions create separate representations for the two different modalities, projecting them to the same shared space and performing a similarity search within that space. Such models are typically based on neural networks and are trained end-to-end [11, 12]. The recently proposed methods exploited semantic category labels to learn discriminative features for cross-modal retrieval [13, 14, 15]. Adversarial learning [16] has also been employed to aid cross-modal retrieval [14, 15]. Further work shows the benefits of applying attention to capture fine-grained relationships between vision and language, creating a better aligned joint embedding space [17].

Cross-modal *recipe* retrieval is a subproblem of the general cross-modal retrieval problem. In this case, one modality is a recipe image, and the second one is a structured text, consisting of a recipe title, a list of ingredients in free form and a list of instructions, also written in free form. It was introduced by Salvador *et al.* [4], who used margin loss for learning the shared embedding space. The image processing pipeline was based on ResNet-50 [18]. Recipe ingredients were normalized using a separate model involving bi-directional LSTM [19] and further encoded with word2vec [20]. The list of instructions was encoded using skip-thoughts [21]. The encoded ingredients and instructions were then passed through to separate LSTMs and concatenated, thus generating the encoding of the recipe text. The resulting model was trained end-to-end (except for word2vec and skip-thoughts vectors which were pretrained separately), improving the top-1 accuracy of **CCA** [5] baseline from 14.0 to 25.6 on a test set of size 1,000 [4]. This model is further referred to as **Pic2Recipe**.

The follow-up work has been largely focused on expanding on the above setup, with the focus on improving cross-modal alignment techniques and minor changes in individual modality processing pipelines and training methods. For example, Chen *et al.* [6, 22] analyzed the importance of instructions and ingredients for cross-modal retrieval and then built a better text representation to match the original metric [7], relying on a Convolutional Neural Network (CNN) pretrained on another food image dataset rather than learning it end-to-end. This model is denoted in this paper as **AM** following [10].

Carvalho *et al.* [8] made improvements to the alignment loss function using double-triplet loss in their **AdaMine** model, improving the top-1 accuracy to 39.8. Zhu *et al.* [9] in their **R2GAN** method employed Generative Adversarial Networks (GANs) [16] to help with learning the representations and reaching results similar to AdaMine. Finally, the current SoTA, **ACME** [10], also uses GANs in addition to a cross-modal triplet loss scheme [15] together with an effective sampling strategy [23], modality alignment using an adversarial learning strategy from [15] and a cross-modal translation consistency loss to reach an impressive 51.8 top-1 accuracy, more than doubling the original performance of Pic2Recipe.

All of the described cross-modal recipe retrieval models reported significant benefits from using a *semantic regularization* technique, where the image and text embeddings are constrained by an additional classification loss with the labels being the categories of the recipes.

On the related topic of building powerful food image classifiers there has been an independent body of work focused around food image datasets [24, 25, 26, 27]. The researchers have explored a variety of architectures more suitable to food images than the ResNet-50 backbone used in cross-modal retrieval [28].

For recipe text encoding, the body of literature is less organized. As there are no commonly used benchmarks for evaluating recipe text representations, the models are usually tuned as part of a bigger task, such as cross-modal retrieval [4], recipe translation [29] or ingredient pairing [30].

Nearest Neighbour Search has been shown to outperform many more complex deep learning methods on neural recommendation tasks [31]. It has also been used successfully for fine-grained image retrieval as a base for many query expansion [32] and database augmentation [33] strategies, which work on top of other methods and yield significant gains on image retrieval tasks[3]. However, to the best of our knowledge, these ideas have not been extended for cross-modal retrieval.

3 Method

Since one of the purposes of this work is to create strong baselines and compare encoders rather than aim for the best possible method of cross-modal recipe retrieval, we do not train our model end-to-end unlike existing approaches [4, 8, 10]. Instead we build a textual recipe encoder (Section 3.1), an image encoder (Section 3.2) and an alignment module (Sections 3.3 and 3.4) independently.

We adopt a widely-used approach for training encoders using a self-supervised classification objective [34, 35], which we setup by automatically extracting noisy labels from recipe titles inspired by [4]. Our encoders employ basic architectures: the last layer of a CNN for images [36, 37] and the average of word embeddings in a bag-of-words for textual recipes [38, 39].

We describe two different modality alignment modules applied on top of precomputed image and text embeddings. In Section 3.3 we propose a novel non-parametric method well-suited for comparing different encoders and creating retrieval baselines. In Section 3.4 we describe a standard triplet loss model optimized for retrieval that achieves SoTA results on image-to-textual-recipe task.

3.1 Bag-of-words encoder (BOW-Encoder)

To build a textual recipe encoder we train a classifier which predicts automatically extracted labels given a textual recipe. To extract a set of C_t noisy labels, we select frequent unigrams and bigrams from recipe titles and filter them by a threshold. Unlike Salvador *et al.* [4], we allow overlapping labels and multiple labels per recipe. Therefore, we use recipe instructions and ingredients as training input data, and titles to extract labels.

We use a standard bag-of-words model as an encoder. The bag-of-words contains instructions and ingredients treated as one long, continuous document. We then add a trainable, randomly initialized word embedding layer with $D_t = 300$ dimensions. The next layer computes the mean of all the embeddings, a linear layer with sigmoid activations is added on top for multi-label classification. Binary cross-entropy is used as a loss function. The architecture is shown in Figure 1.

[3]https://landmarksworkshop.github.io/CVPRW2019/

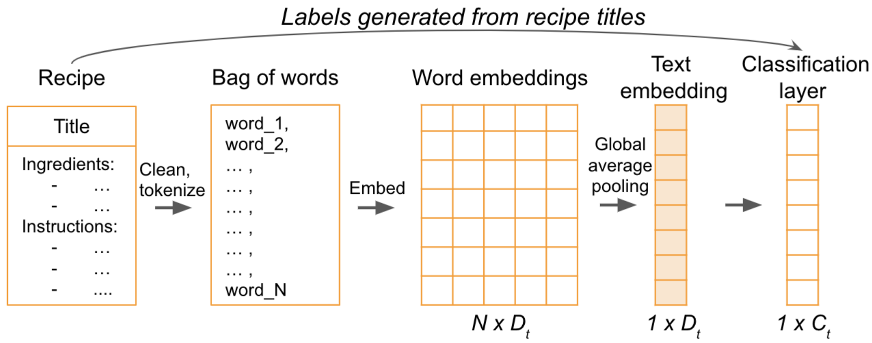

Figure 1: Pipeline for the Bag-of-Words Encoder.

When inferring the text embedding for the recipe, we take the average of word embeddings for the document. To include the information from the recipe title into the text embedding, we add the recipe title words to the bag-of-words at inference time. We refer to this model as *BOW-Encoder*.

3.2 Recipe image encoders (ResNet- and ResNext-Encoder)

For creating a recipe image encoder we train a multi-label classifier on recipe images. Because the input to this model is images, we expand the set of labels used for training BOW-Encoder with additional frequent unigrams and bigrams from the ingredients list. Therefore, we use the combined set of noisy labels extracted from titles and ingredient lists to train our image encoder.

We use ResNet-50 [18] as a CNN architecture, and apply binary cross-entropy loss for multi-label classification. We have also retrained the same model with ResNext-101 architecture [40]. We refer to these models as *ResNet-Encoder* and *ResNext-Encoder* respectively. To create recipe image embeddings, we extract features from the last convolutional layer of the network after global average pooling.

3.3 Cross-Modal k-Nearest-Neighbours (CkNN)

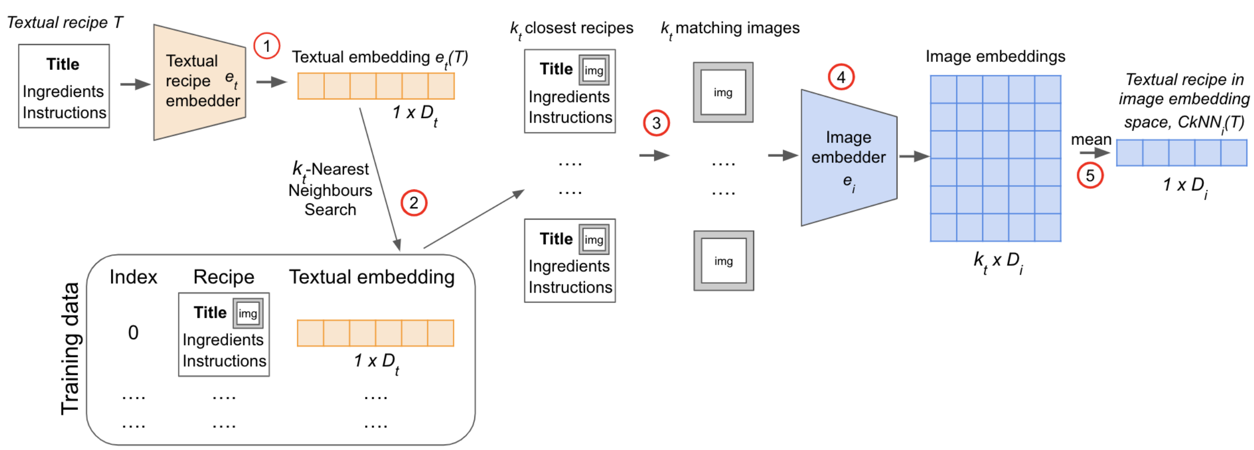

Figure 2: A schematic representation of the core idea of our CkNN method: representing a textual recipe T in the image embedding space using nearest neighbour search in the training data. Orange and blue shapes respectively denote text and image modalities. Numbers in red circles correspond to $\mathrm{CkNN}_i(T)$ algorithm steps described in Section 3.3

Our non-parametric CkNN method belongs to the category of alignment modules, which attempt to match the representations of different modalities. CkNN is applied on top of a pretrained image encoder e_i with a distance measure d_i in the image embedding space and a pretrained textual recipe encoder e_t with a distance measure d_t in the text embedding space. In practice, we use cosine distance in the embedding space as the distance measures d_i and d_t, although they could be different from each other in principle.

CkNN uses the training data to represent a candidate textual recipe T in the image embedding space using the following algorithm denoted as $\mathrm{CkNN}_i(T)$ and also depicted in Figure 2:

1. Given a candidate textual recipe T, encode it in the text embedding space as $e_t(T)$.
2. Find the set $\mathcal{R}_T$ of k_t nearest neighbours of T in the text embedding space in the database of training recipes, according to the distance measure d_t.
3. Extract the set of images $\mathcal{I}_T$ corresponding to the set of selected closest recipes $\mathcal{R}_T$.
4. Encode each $I \in \mathcal{I}_T$ in the image embedding space as $e_i(I)$ (this step could be precomputed, as the images belong to the training set and are known in advance).
5. Return the mean vector $\frac{1}{|\mathcal{I}_T|}\sum_{I\in\mathcal{I}_T} e_i(I)$.

Similarly, the mirror $\mathrm{CkNN}_t(I)$ algorithm represents a query image I in the text embedding space. This algorithm uses a different parameter k_i for the nearest neighbour search.

Thus, we now have two ways to calculate the distance between a query image I and a candidate textual recipe T: one in the image embedding space, and one in the text embedding space. We therefore define the total distance d_{CkNN} as a linear combination of the two with a parameter $0 \le \alpha \le 1$ as illustrated in Figure 3 and given by Eq. (1). We can then use d_{CkNN} to rank the set of candidate textual recipes given a query image.

$$d_{\mathrm{CkNN}}(I,T) = \alpha\, d_i(e_i(I), \mathrm{CkNN}_i(T)) + (1-\alpha)\, d_t(\mathrm{CkNN}_t(I), e_t(T)) \tag{1}$$

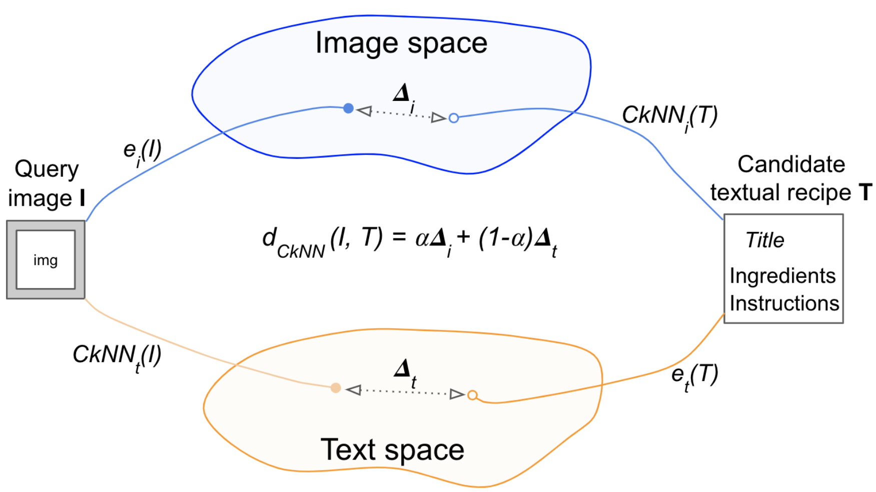

Figure 3: CkNN distance $d_{\mathrm{CkNN}}(I,T)$ between a query image I and a candidate textual recipe T to be used in ranking for the cross-modal retrieval task.

Note, that the embeddings computed using the encoders from Sections 3.2 and 3.1 are not designed to be used for retrieval using cosine similarity. We refrain from applying advanced methods such as ArcFace [41] for building encoders to keep our paper focused on baselines and standard approaches for representation learning.

3.4 Triplet loss alignment module

Our second approach of aligning the modalities is based on the standard triplet loss [42]. This method is still applied on top of precomputed embeddings, but these embeddings are now re-projected to a common space. Namely, we jointly train two feed-forward neural networks with one hidden layer, dropout and batch normalization: one for image (g_i) and another for textual (g_t) features with triplet loss. Architectures of both neural networks are identical, and output feature size is $D = 1024$. This pipeline is depicted in Figure 4.

Each triplet consists of one feature embedding as an anchor point in image modality and a positive and negative feature embeddings from text modality. The positive instances are the different modalities of the same recipe, X_{ia} for image and X_{ta} for textual features. We use online negative instance mining [42] to choose the negative instance X_{tn} as the closest text instance to the anchor point selected from other recipes in the mini-batch. The objective $\mathcal{L}$ is given by Eq. (2).

$$\mathcal{L} = \max(0, d(g_i(X_{ia}), g_t(X_{ta})) - d(g_i(X_{ia}), g_t(X_{tn})) + \gamma), \tag{2}$$

where d is the cosine distance between two vectors and γ is the margin.

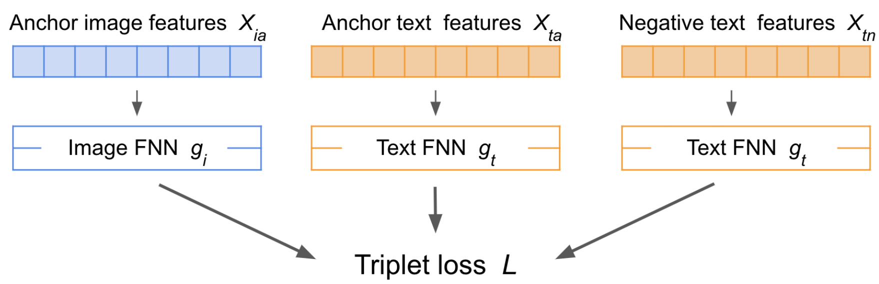

Figure 4: Projecting the precomputed image and textual features in the same common space using triplet loss. X_{ia} has D_i dimensions, X_{ta} and X_{tn} have D_t dimensions

4 Experiments and Results

4.1 Dataset and metrics

The Recipe1M dataset [4] consists of over 1M textual cooking recipes (including recipe title, list of ingredients in free form, and list of cooking instructions). In addition, 402,760 of those recipes are linked to one or more corresponding images (887,706 images in total). The dataset is split into dedicated training, validation and test sets.

In our experiments we use the standard metrics on image-to-textual-recipe retrieval task for the Recipe1M dataset as proposed by [4]:

- Recall at Top K (R@K) for $K = 1, 5, 10$ describes the percentage of images for which the correct textual recipe is among the top K of the ranked results list (higher is better).
- Median Rank (medR) is the median of the rank of the correct textual recipes across all images (lower is better).

The cross-modal retrieval metrics are calculated in the same way as in the prior work. Namely, we randomly sample $N = 1,000$ (1K setup) or $N = 10,000$ (10K setup) recipes from the test set, and for each test image, ranking the textual recipes in the sample using the given model.

The metrics above are noisy due to randomization, so we follow the strategy proposed in [4] and sample the sets 10 times, reporting the average results. It should be noted that the results are still too noisy in the case of $N = 1,000$ due to sampling errors [8], thus in this work we focus on metrics for $N = 10,000$ (although we also report the $N = 1,000$ performance for comparison against previously published results in Section 4.3).

To focus on one metric for clarity in our evaluation section, we select the recall at rank-1 with the sample size of 10,000 (**10K-R@1**), but we note that the results are consistent for all of the metrics. Throughout the paper, we only use the Recipe1M validation set for validation purposes, and report the results on the test set.

4.2 Comparison of image and text models for cross-modal recipe retrieval

Since CkNN modality alignment fully decouples recipe image and text encoders and depends directly on image and text distance measures d_i and d_t (Eq. 1), the performance on cross-modal retrieval could indicate how good the individual modality embeddings are for retrieval purposes. We thus compare encoders pretrained with different methods by applying CkNN to all combinations of image and text encoders and report 10K-R@1 metric on the Recipe1M test set[4]. For example, we take the image encoder trained jointly as part of SoTA ACME [10] model and use it in combination with text encoder trained jointly as part of Pic2Recipe [4].

[4]Here we use only such training recipes for CkNN for which the embeddings are available for all the encoders being compared. This accounts for the small discrepancy with numbers reported in Section 4.3.

Image Model \ Text Model	ACME* [10]	Pic2Recipe [4]	TF-IDF	BOW-Encoder
ACME* [10]	17.9	13.3	10.6	15.6
Pic2Recipe [4]	8.5	7.1	5.3	7.5
AdaMine [8]	12.6	9.9	7.5	11.2
ImageNet-Pretrained	4.4	3.4	3.8	5.0
Food-Pretrained	7.8	6.2	6.5	8.6
ResNet-Encoder	16.6	13.1	12.0	17.4

Table 1: 10K-R@1 metric on the Recipe1M image-to-textual-recipe task, computed for various image and text encoders combined using our CkNN approach (higher is better). All the image models are built on ResNet-50 [18] backbone. Although the results are not optimal, they are competitive with published methods [10], and allow for direct comparisons between different image and textual recipe embeddings.

We have manually tuned the hyperparameters of CkNN on 10K-R@1 metric on the Recipe1M validation set to $\alpha = 0.1,\ k_t = 15,\ k_i = 3$. We found this set of hyperparameters to work well for all the image and text encoders we compare. This, along with the fact that CkNN does not require training, also contributes to CkNN being a suitable choice for comparing encoders.

4.2.1 Image models

For comparison purposes, we used the existing image encoders trained for cross-modal recipe retrieval tasks, which are described in detail in Section 2. We used only the models for which we could run the inference code, which are **Pic2Recipe** [4] and **AdaMine** [8]. It should be noted that we could not reproduce the published performance for ACME [10] and our metrics for this model are slightly worse, as can be seen in Table 2. We thus refer to this model as **ACME***.

We further used the following baseline image encoders pretrained on different public domain datasets to extract the embeddings from their last convolutional layer: **ImageNet-Pretrained** pretrained on ImageNet [43] and **Food-Pretrained** pretrained on the concatenation of Food-101 [24], ChineseFoodNet [25] and iFood-2018[5] datasets.

We also used **ResNet-Encoder** method described in Section 3.2. We applied it to the Recipe1M training set and extracted 5036 labels for 281,558 recipes with an average of 10.7 labels per recipe. We train the model for 40 epochs with Adam optimizer [44], a batch size of 512 and an initial learning rate of 0.0001. All the models in this section are based on the ResNet-50 backbone [18].

4.2.2 Text models

The only two publicly available textual recipe encoders that we could run at the moment of writing are **ACME*** and **Pic2Recipe**, described in detail in Section 2.

As a baseline unsupervised encoder we represented the textual recipes as a bag-of-subword-units with term frequency-inverse document frequency (TF-IDF) weights calculated on the Recipe1M training set. We applied singular value decomposition on top of this representation, reducing the dimensionality of the embedding to $D_t = 2000$. The subword units were selected as in FastText [45]. The model is denoted as **TF-IDF**.

We also compared the above models with **BOW-Encoder** model introduced in Section 3.1. We extract $C_t = 3453$ labels for 679,105 recipes from the Recipe1M training set with 2.5 labels per recipe on average. We train the model with Adam optimizer [44] and an initial learning rate of 0.002 for 15 epochs.

4.2.3 Comparison of previously published model components

The results of our experiments are shown in Table 1. The first observation is that the performance of CkNN is competitive with direct search in the embedding space for which the jointly trained encoders were optimized [4, 10]. Indeed, combining ACME* image and text encoders using CkNN drops the performance of direct search only from 20.6 to 17.9, which is still better than all previously published methods. For Pic2Recipe, the drop is even smaller: from 7.3 to 7.1.

We further observe that the performance of individual encoders, image or text, is generally consistent across different combinations and in line with performance on the Recipe1M dataset as reported in [10], validating our comparison

[5] https://github.com/karansikka1/Foodx

framework using CkNN. ACME* (SoTA method) image encoder produces consistently higher numbers across all four text encoders than AdaMine (2nd best method) image encoder, and AdaMine is better than Pic2Recipe (3rd best method) across all metrics. ACME* text encoder outperforms Pic2Recipe text encoder across all 6 image encoders.

The two observations above indicate that modality alignment procedures used in the existing approaches add on the order of 10% improvement to 10K-R@1 metric compared to CkNN. At the same time, the performance differences due to replacing the encoders are around 50% from the 3rd to the 2nd best result, and from the 2nd best result to the SoTA result. This suggests that the large performance gap in the reported performance of existing cross-modal methods could be explained primarily by the strengths of the learned image and text embedding spaces for retrieval purposes, and not by the quality of cross-modal alignment.

4.2.4 Comparison of independently trained encoders

We now compare the results obtained with the encoders trained jointly as part of the existing cross-modal methods and other text and image encoders, trained independently. We observe that the image encoders pretrained on external data combined with unsupervised TF-IDF encoder produce results on a par with some published metrics. Indeed, Food-Pretrained image encoder outperforms Pic2Recipe image encoder in combination with some of the text encoders, and its combination with TF-IDF scores close to direct search with Pic2Recipe. Even Imagenet-Pretrained with TF-IDF produces a reasonable score of 3.8 (a random model would yield 10K-R@1 score of 0.01)[6]. This shows that CkNN allows easy creation of many competitive baselines out of encoders trained in completely different ways.

Next, we analyze the performance of our proposed self-supervised encoders. Among image encoders, ResNet-Encoder performs close to ACME* and consistently outperforms other encoders. Among text encoders, BOW-Encoder reaches performance similar to ACME*, whereas Pic2Recipe and TF-IDF perform much worse. This suggests that independent training using a self-supervised classification objective can produce encoders competitive with those trained as part of the SoTA cross-modal retrieval methods. In addition, the success of a much simpler bag-of-words architecture compared to the complex ACME* and Pic2Recipe textual model architectures suggests that more research is needed to understand how to best represent textual recipes.

4.3 Improving the baseline to reach SoTA

While ACME* image and text embeddings combination outperforms all others, we note that ResNet-Encoder and BOW-Encoder score is very close. In fact, CkNN combined with the embeddings precomputed using these encoders (*CkNN+BOW+ResNet*) provides a strong cross-modal recipe retrieval baseline that improves upon a second-best published result [9]. In this section, we show how to improve *CkNN+BOW+ResNet* to achieve new SoTA, summarizing our results in Table 2.

4.3.1 Triplet loss alignment module

Although we showed that CkNN is a suitable choice for model comparison and building cross-modal retrieval baselines, it is by no means the optimal method for aligning the modalities. As observed in Section 4.2, direct search through a jointly learned embedding space can surpass the results of our CkNN approach for ACME* and Pic2Recipe encoders. Thus, we also train a triplet loss alignment module to create a joint embedding space on top of the precomputed image and text embeddings as described in Section 3.4. This boosts 10K-R@1 metric to 26.5, yielding a new SoTA result (*Triplet+BOW+ResNet* in Table 2).

The model was trained on 238,408 recipes from the Recipe1M training set, with the margin γ manually tuned to be 0.3 on 10K-R@1 metric on a validation set. We used Adam optimizer [44], a batch size of 256, initial learning rate of 0.002, and applied alternating optimization [4] to aid convergence. Training takes only 25 seconds per epoch on a Tesla M60 GPU.

We emphasize that these SoTA results were achieved by applying a small alignment module on top of features precomputed from an independently trained ResNet-50 image encoder and a bag-of-words textual recipe encoder. This is in contrast to the complexity of the previous SoTA approach, which jointly trained image and text models using GANs; an adversarial alignment module; a novel hard negative mining strategy; translation consistency losses; classification losses; and multiple bidirectional LSTMs on top of skip-thought vectors and dedicated ingredient embeddings for textual recipe representation [10]. This suggests that the retrieval metrics on the Recipe1M dataset still have a lot of room for improvement, and we expect large gains to be achieved with advanced end-to-end methods in future work.

[6] This combination also achieves 1K-R@1 of 15.2 on 1K test set, outperforming CCA baseline of 14.0 reported by [4]

Size of test set	Method	medR ↓	R@1 ↑	R@5 ↑	R@10 ↑
1K	CCA [5]	15.7	14.0	32.0	43.0
	Pic2Recipe [4]	5.2	25.6	51.0	65.0
	AM [7]	4.6	25.6	53.7	66.9
	AdaMine [8]	2.0	39.8	69.0	77.4
	R2GAN [9]	2.0	39.1	71.0	81.7
	(Previous SoTA) ACME [10]	1.0	51.8	80.2	87.5
	ACME* [10]	1.8	49.0	77.1	85.2
	(Ours) CkNN+BOW+ResNet	2.0	45.7	75.9	84.2
	(Ours) CkNN+BOW+ResNext	1.3	50.5	79.5	86.7
	(Ours) Triplet+BOW+ResNet	1.0	55.9	82.4	88.7
	(Ours) Triplet+BOW+ResNext	**1.0**	**60.2**	**84.0**	**89.7**
10K	Pic2Recipe* [4]	39	7.3	20.3	29.0
	AM [7]	39.8	7.2	19.2	27.6
	AdaMine [8]	13.2	14.9	35.3	45.2
	R2GAN [9]	13.9	13.5	33.5	44.9
	(Previous SoTA) ACME [10]	6.7	22.9	46.8	57.9
	ACME* [10]	7.5	20.6	44.3	55.7
	(Ours) CkNN+BOW+ResNet	9.1	19.1	41.3	52.5
	(Ours) CkNN+BOW+ResNext	6.8	22.9	46.9	58.0
	(Ours) Triplet+BOW+ResNet	5.0	26.5	51.8	62.6
	(Ours) Triplet+BOW+ResNext	**4.0**	**30.0**	**56.5**	**67.0**

Table 2: Performance of our method and other reported methods on Recipe1M image-to-textual-recipe task. The best results are shown in bold, and are statistically significant. The description of the existing methods could be found in Section 2. The numbers computed by us using publicly available models are denoted by *. While we report the results for both 1K and 10K test sample size, we only rely on 10K values for our analysis since 1K results are too noisy [8].

Method	medR ↓	R@1 ↑	R@5 ↑	R@10 ↑
(Previous SoTA) ACME [10]	6.7	22.9	46.8	57.9
(Ours) †Triplet+BOW+ResNet	5.9	24.4	49.4	60.5
(Ours) Triplet+BOW+ResNet	5.0	26.5	51.8	62.6
(Ours) †Triplet+BOW+ResNext	4.0	28.6	54.8	65.6
(Ours) Triplet+BOW+ResNext	**4.0**	**30.0**	**56.5**	**67.0**

Table 3: Ablation study of the new SoTA results. † symbol next to the model means using only training data available to ACME [10] method after preprocessing and selecting recipes paired with images. We only report on 10K test set since 1K results are too noisy [8].

4.3.2 Increasing the capacity of the image encoder

Since our ResNet-Encoder is trained using a classification objective, it is an obvious extension to replace ResNet-50 with ResNext-101 [40] architecture, which performs better on standard classification benchmarks such as ImageNet [43].

When we use ResNext-Encoder, 10K-R@1 metric for CkNN (*CkNN+BOW+ResNext*) reaches 22.9, matching previous SoTA results from [10], and with triplet loss (*Triplet+BOW+ResNext*) it is boosted to 30.0, which further improves on previous SoTA by a large margin. It remains to be seen to what extent the existing methods would benefit from other image architectures.

In addition, though in this paper we only focus on image-to-textual-recipe retrieval task for clarity and the ease of analysis, we also report our best model's performance on the mirror textual-recipe-to-image task for completeness. The proposed method achieves 10K-medR of 4.0, 10K-R@1 of 30.5, 10K-R@5 of 56.3 and 10K-R@10 of 66.6, which also improves upon the previous SoTA results by a large margin [10].

4.3.3 Ablation study: impact of training data utilization

One benefit of training text and image models separately as described in Section 3.1 and 3.2, is that one can make use of training data which was filtered out by the existing methods' preprocessing pipeline [4, 10]. Indeed, since the setup for training BOW-Encoder does not require any image-to-textual-recipe pairs and is entirely self-supervised, we are able to train it on 680K textual recipes from the Recipe1M training set, as opposed to only 240K recipes with images filtered by ACME for joint training [10]. Similarly, we utilize 280K recipes with images for ResNet-Encoder/ResNext-Encoder compared to 240K filtered by ACME [10]. It should be noted, however, that ingredient and instruction embeddings from Pic2Recipe and ACME were also trained on the full 1M dataset [4].

To see by how much our best models' performance was improved by better training data utilization, we train our encoders only on the subset of images and recipes used by ACME [10] for joint training. Triplet loss model has already been using the same subset. We observe that 10K-R@1 metric dropped from 30.0 to 28.6 for ResNext, and from 26.5 to 24.4 for ResNet, which is a sizeable difference but is not fundamental for the performance of our method (†*Triplet+BOW+ResNet* and †*Triplet+BOW+ResNext* in Table 3). These results still improve previous SoTA.

5 Conclusions and Future Work

Our work shows that all the existing methods for cross-modal recipe retrieval could be outperformed with unsophisticated approaches, echoing the sentiment by Dacrema *et al.* [31] from neural recommendation. We hope that our comparison framework would help researchers to improve individual modules of advanced jointly trained methods, while the presented baseline methods and SoTA results would move the goalposts to aid further progress. As the methods are not specific to recipe retrieval, we hope that this work would aid similar analysis for other cross-modal domains.

References

[1] Javier Marín, Aritro Biswas, Ferda Ofli, Nicholas Hynes, Amaia Salvador, Yusuf Aytar, Ingmar Weber, and Antonio Torralba. Recipe1m: A dataset for learning cross-modal embeddings for cooking recipes and food images. *arXiv preprint arXiv:1810.06553*, abs/1810.06553, 2018.

[2] Austin Myers, Nick Johnston, Vivek Rathod, Anoop Korattikara, Alex Gorban, Nathan Silberman, Sergio Guadarrama, George Papandreou, Jonathan Huang, and Kevin Murphy. Im2Calories: Towards an Automated Mobile Vision Food Diary. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1233–1241. IEEE, 2015.

[3] Jill Freyne and Shlomo Berkovsky. Recommending food: Reasoning on recipes and ingredients. In *User Modeling, Adaptation, and Personalization*, pages 381–386, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[4] Amaia Salvador, Nicholas Hynes, Yusuf Aytar, Javier Marin, Ferda Ofli, Ingmar Weber, and Antonio Torralba. Learning cross-modal embeddings for cooking recipes and food images. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3068–3076, July 2017.

[5] Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3-4):321–377, 12 1936.

[6] Jing-jing Chen, Chong-Wah Ngo, and Tat-Seng Chua. Cross-modal recipe retrieval with rich food attributes. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, pages 1771–1779, New York, NY, USA, 2017. ACM.

[7] Jing-Jing Chen, Chong-Wah Ngo, Fu-Li Feng, and Tat-Seng Chua. Deep understanding of cooking procedure for cross-modal recipe retrieval. In *Proceedings of the 26th ACM International Conference on Multimedia*, MM '18, pages 1020–1028, New York, NY, USA, 2018. ACM.

[8] Micael Carvalho, Rémi Cadène, David Picard, Laure Soulier, Nicolas Thome, and Matthieu Cord. Cross-modal retrieval in the cooking context: Learning semantic text-image embeddings. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pages 35–44, New York, NY, USA, 2018. ACM.

[9] Bin Zhu, Chong-Wah Ngo, Jingjing Chen, and Yanbin Hao. R2GAN: Cross-modal recipe retrieval with generative adversarial network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[10] Hao Wang, Doyen Sahoo, Chenghao Liu, Ee-peng Lim, and Steven C. H. Hoi. Learning cross-modal embeddings with adversarial networks for cooking recipes and food images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11572–11581, 2019.

[11] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. Cross-modal retrieval with correspondence autoencoder. In *Proceedings of the 22Nd ACM International Conference on Multimedia*, MM '14, pages 7–16, New York, NY, USA, 2014. ACM.

[12] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pages 689–696, USA, 2011. Omnipress.

[13] Yuxin Peng, Jinwei Qi, Xin Huang, and Yuxin Yuan. CCL: Cross-modal correlation learning with multigrained fusion by hierarchical network. *IEEE Transactions on Multimedia*, PP, 04 2017.

[14] Yuxin Peng, Jinwei Qi, and Yuxin Yuan. Cm-gans: Cross-modal generative adversarial networks for common representation learning. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 15, 10 2017.

[15] Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen. Adversarial cross-modal retrieval. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, pages 154–162, New York, NY, USA, 2017. ACM.

[16] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

[17] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 2016.

[19] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 3111–3119, USA, 2013. Curran Associates Inc.

[21] Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Skip-thought vectors. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, pages 3294–3302, Cambridge, MA, USA, 2015. MIT Press.

[22] Jingjing Chen, Lei Pang, and Chong-Wah Ngo. Cross-modal recipe retrieval: How to cook this dish? In *MultiMedia Modeling*, pages 588–600, Cham, 2017. Springer International Publishing.

[23] Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, June 2015.

[24] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014.

[25] Xin Chen, Hua Zhou, and Liang Diao. ChineseFoodNet: A large-scale image dataset for Chinese food recognition. *arXiv preprint arXiv:1705.02743*, 2017.

[26] Hamid Hassannejad, Guido Matrella, Paolo Ciampolini, Ilaria De Munari, Monica Mordonini, and Stefano Cagnoni. Food image recognition using very deep convolutional networks. In *Proceedings of the 2Nd International Workshop on Multimedia Assisted Dietary Management*, MADiMa '16, pages 41–49, New York, NY, USA, 2016. ACM.

[27] Ashutosh Singla, Lin Yuan, and Touradj Ebrahimi. Food/non-food image classification and food categorization using pre-trained GoogLeNet model. In *Proceedings of the 2Nd International Workshop on Multimedia Assisted Dietary Management*, MADiMa '16, pages 3–11, New York, NY, USA, 2016. ACM.

[28] Niki Martinel, Gian Luca Foresti, and Christian Micheloni. Wide-slice residual networks for food recognition. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 567–576, March 2018.

[29] Takayuki Sato, Jun Harashima, and Mamoru Komachi. Japanese-English machine translation of recipe texts. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pages 58–67, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[30] Donghyeon Park, Keonwoo Kim, Yonggyu Park, Jungwoon Shin, and Jaewoo Kang. Kitchenette: Predicting and ranking food ingredient pairings using siamese neural network. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5930–5936. International Joint Conferences on Artificial Intelligence Organization, 7 2019.

[31] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches. Proceedings of the 13th ACM Conference on Recommender Systems (RecSys 2019), 2019.

[32] Ondrej Chum, James Philbin, Josef Sivic, Michael Isard, and Andrew Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, Oct 2007.

[33] Panu Turcot and David Lowe. Better matching with fewer features: The selection of useful features in large database recognition problems. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pages 2109–2116, Sep. 2009.

[34] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arxiv:1810.04805*, 2018.

[35] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pages 1422–1430, Washington, DC, USA, 2015. IEEE Computer Society.

[36] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky. Neural codes for image retrieval. In *Computer Vision – ECCV 2014*, pages 584–599, Cham, 2014. Springer International Publishing.

[37] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 06 2015.

[38] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. *31st International Conference on Machine Learning, ICML 2014*, 4, 05 2014.

[39] Felix Hill, Kyunghyun Cho, and Anna Korhonen. Learning distributed representations of sentences from unlabelled data. In *HLT-NAACL*, 2016.

[40] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *arXiv preprint arXiv:1611.05431*, 2016.

[41] Jiankang Deng, Jia Guo, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. *arXiv preprint arXiv:1801.07698*, 2018.

[42] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[43] Jia Deng, Wei Dong, Richard Socher, Li-Joa Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[44] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[45] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.